

Using Agent-Based Modeling in the Exploration of Self-Organizing Neural Networks

Timothy Schoenharl

Greg Madey

Department of Computer Science and Engineering

University of Notre Dame

Notre Dame, IN 46556 USA

Abstract

In this paper we leverage the power of agent-based modeling to explore a novel self-organizing neural network topology. We have drawn inspiration from recent research into complex networks and advances in neurobiology, and applied it towards the construction of a neural network. Techniques from agent-based modeling are used to simplify the construction process and provide flexibility in modifying the simulation. The experiment also implements ideas from swarm programming, using local information to develop global structure. We demonstrate our simulation, modeled using the RePast framework.

1 Introduction

1.1 Complex Networks

In recent years there have been significant developments in the analysis of complex networks. The results of Watts and Strogatz [1] and Barabási *et al* [2][3] are well-known and often cited. Researchers have discovered that many complex networks, such as the World Wide Web [3], the neural network of the nematode *C. Elegans* [1] and the actor collaboration network [4][5] share important characteristics. These networks do not have a regular structure; however, they are not completely random either [1].

Watts and Strogatz, in [1], analyze the actors collaboration network, the neural network of *C. Elegans* and the power grid of the western United States. They show that these diverse networks display similar characteristics, namely short characteristic path length and high average clustering coefficients. Characteristic path length can be described simply as “the typical distance between every vertex and every other vertex” [4]. The average clustering coefficient refers to the average level of interconnectedness of the neighborhood around each node [4]. For a more formal def-

inition of these terms, readers are directed to [4].

Graphs displaying both high clustering coefficients and short characteristic path length have been referred to as “small-world” graphs. Some of the benefits of small world graphs include “enhanced signal-propagation speed [and] computational power” [1]. Additionally, high clustering coefficient in a neural network can be interpreted as a large number of lateral connections. According to [6], lateral connections in a neural network can aid in pattern enhancement.

Barabási *et al* have analyzed numerous natural and artificial networks, including cellular metabolic networks [7], the actor collaboration network [5] and the power grid of the western United States [5]. Barabási *et al* highlight a characteristic that differentiate these graphs from random graphs, a “scale-free” link distribution [2]. Networks with a “scale-free” link distribution have a small diameter and have a high tolerance for random node failure [8]. Diameter is defined as “the average length of the shortest path between any two nodes in the network” [8]. A tolerance for random failures is important in natural neural networks, since neurons do not regenerate like other cells [9].

It is important to note that Barabási *et al* describe an algorithm for constructing a “scale-free” network. This algorithm relies on growing node by node with preferential attachment [5]. We take Barabási’s algorithm as inspiration for our network organization method, but we modify it so that only local information is used.

1.2 Artificial Neural Networks

Artificial neural network research began in 1943 with the work of McCulloch and Pitts [10](as cited in [6]). In subsequent years, a major focus of artificial neural network (ANN) research has been on the development of successively more accurate models of biological neurons [11]. One of the most basic designs is the perceptron, a more advanced design is the “spiking”

neuron [11]. This design more accurately captures the behavior of biological neurons, which can lead to more powerful artificial neural networks [12].

There has also been significant work done on the topology of artificial neural networks. Current designs range from feed-forward networks to more complex topologies such as Adaptive Resonance Theory (ART) [9]. Most designs of neural network require that the network topology be defined off line, before training begins. It is known that the topology affects the computational ability of the ANN [9], thus it is vital that the ANN have a suitable topology. There is a class of ANN that develops its own topology, the self-organizing map (SOM) [9]. These maps do self-organize, but the mechanism involved in their organization is not considered to be biologically plausible [9], suggesting that there is room for alternate designs.

1.3 Biological Neural Networks

According to [13], human DNA lacks the storage capability to contain an exact plan of the wiring of the human brain. Thus on some scale, the brain must self-organize. Although the large-scale (above 1cm) structure of the brain is largely deterministic, the small-scale (below 1mm) structure of the brain appears to be random [13]. However, given the human capacity for knowledge and memory, it is obvious that the human neural network must possess some form of organization.

In the early stages of development, from before birth until puberty, the mammalian brain is subjected to a massive loss of neurons, axons and synapses [14]. Approximately half of the neurons and about one third of the synapses will die off before adolescence. The neurons, axons and synapses do not die off randomly, instead it appears that the selection process is related to Hebbian learning [14]. Chechik and Meilijson [15] suggest that *neuronal regulation*, a biological mechanism that maintains post-synaptic membrane potential, may play a part in the synaptic pruning process.

1.4 Agent Based Modeling, Swarm Simulation

This simulation differs from the computation neural networks described above, as well as the simulations used by neurobiologists to study neurobiological mechanisms. By modeling neurons as agents, we can add more complex behaviors in addition to response functions. The neurons in our simulation conduct evaluation of their links and control the pruning

process based on local criteria. The interaction of the neurons through signaling enables evaluation of fitness, allow neurons to make decisions affecting global structure. The neural network topology thus emerges as the result of these local interactions. Central to this emergence is the utilization of feedback.

An agent based approach offers gains in terms of modeling neuron complexity over other approaches, such as stochastic simulations. Our simulation utilizes the RePast framework, which has a sophisticated discrete time event simulator [19] and several network analysis tools. It would be difficult to get the same functionality from neural net simulation written purely in Java. This would require the replication of a scheduling mechanism, if not an entire library of support code. Utilizing an ABM framework allows us to concentrate on details that are important to our investigation. RePast is written entirely in Java, allowing us to develop and deploy the simulation on heterogeneous hardware (PowerPC, SPARC, Intel) and operating systems (MacOS X, Solaris, Linux).

2 Simulation

2.1 Hebbian Learning and Neuronal Regulation

Hebb's postulate [16], as cited in [17], states:

When an axon of cell A is near enough to excite cell B or repeatedly or persistently takes part in firing it, some growth process or metabolic change takes place in one or both cells such that A 's efficiency, as one of the cells firing B , is increased.

Hebbian learning can be seen as a major mechanism in the selection of neurons and synapses during the early phase of development in biological neural networks [14]. See Figure 1 for a simple description of Hebbian learning. The Hebbian reinforcement of the connections between neurons may be mitigated by neuronal regulation [15][18]. Neuronal regulation, according to [15], is a biological process that acts to "maintain the homeostasis of the neuron's membrane potential." It is a method of maintaining the input level into a neuron in the event of a change in synaptic strength. Thus neuronal regulation can help to bound the effects of Hebbian learning.

In our simulation, we implement a model of Hebbian learning as described in [17], as well as the model of neuronal regulation as described in [15]. In [17], the author describes a model of Hebbian learning for use in rate-based ANNs. We have interpreted the author's

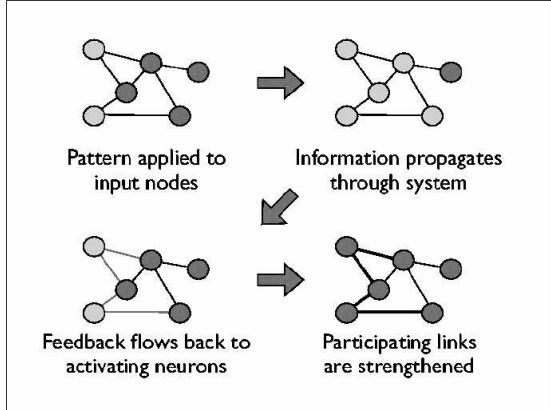


Figure 1: A simplified diagram of the Hebbian learning process.

important equations in terms of our model, which uses simple perceptrons. The equation we use for the adjustment of link strength is as follows:

$$\Delta(w_{ij}) = -\gamma w_{ij}(1 - w_{ij})(w_{\theta} - w_{ij}) \quad (1)$$

with $\gamma > 0$ and $0 < w_{\theta} < 1$. The w_{ij} term is the link weight of the synapse from neuron i to neuron j . The γ term is a scaling factor that varies the intensity of the adjustment. For values of $w_{ij} < w_{\theta}$ the overall adjustment is negative, and for values of $w_{ij} > w_{\theta}$ the overall adjustment is positive. This equation leads to a long term stability in learning, preventing learned patterns from being overwritten [17].

Our model of neuronal regulation comes from [15]. We utilize Chechik’s method of first degrading synaptic weights and then multiplicatively strengthening them. We start with a post-synaptic neuron j , degrading all of its incoming synaptic weights. The degraded weights are determined by the following function:

$$w'_{ij} = w_{ij} - (w_{ij})^{\alpha} \eta \quad (2)$$

Where w is the synaptic strength, η is a Gaussian distributed noise term with positive mean, and α is the *degradation dimension*, $0 \leq \alpha \leq 1$. After applying the degradation function we apply the neuronal regulation function:

$$w_{ij} = w'_{ij} \frac{f_j^0}{f_j^t} \quad (3)$$

Here f_j^t is the input field of neuron j at time t . The term “input field” refers to the sum of all incoming synaptic weights. The neuronal regulation function

maintains the neuron’s input field at its original ($t = 0$) level.

2.2 Neuron Design

We model our neurons with a simple threshold response behavior. As our model is concerned with topology formation related to adjustments in link strength, we do not adjust the response function of the neuron. The neurons are modeled as agents in our system, having their own behaviors and access only to local information. When a neuron receives a pulse from a pre-synaptic neuron, it stores the information about the pulse (arrival time, intensity and originating neuron) and determines whether, with the addition of this pulse, it has exceeded its threshold. If it has exceeded its threshold, it schedules a fire action, schedules a feedback action and then schedules a refraction action. The neuron implements a sliding time window so that pulses received in the distant past do not count towards its decision to fire. We implement the Chechik and Meilijson model [15] of neuronal regulation, which alternately degrades and strengthens the link strength of all incoming synapses into the triggered neuron. This neuronal regulation occurs at every integer time step, regardless of the behavior of the neuron.

When the neuron implements the fire action, it sends a signal to each of its post-synaptic neurons via the shared synapses. The signal consists of the current time in the simulation and the intensity of the pulse. Each intermediary synapse records this information and then passes on the pulse to the post-synaptic neuron. The intensity of a neuron’s pulse is fixed at the start of the simulation and from there on it remains constant.

After firing, a neuron triggers a feedback action. In the feedback action, all of the pre-synaptic neurons that contributed to the firing of the neuron receive a message. The incoming synapses are strengthened in accordance with the principle of Hebbian learning. Pre-synaptic neurons that did not contribute to the firing of this neuron receive no message, thus their associated synapses receive no increase in strength. Thus neuronal regulation and Hebbian learning work together to strengthen useful synapses without creating a runaway effect. The cooperative behavior of these mechanisms is described in [15] and [18].

Upon deciding to schedule a fire action, the neuron enters a refractory period, where it cannot fire. The refractory period is inspired by a biological mechanism [9], and here serves an additional role of limiting the possibility of infinite cycles of pulses. The neuron schedules itself to exit the refractory period

after a certain amount of time. The length of a neuron’s refractory period is set at the beginning of the simulation and is maintained throughout the simulation.

2.3 Initial Setup

We start by creating 100 neurons and randomly distributing them in a 500 by 500 grid. (Side note: We currently do not use location information, as neurons only signal via their synapses. It is known that neurons also signal through the release of chemical signals into the medium, thus in future iterations of the simulation we may want to enable location based signaling.) The neurons are all initialized with same threshold value, though this could easily be modified to be a random value. We iterate through the list of neurons, at each neuron we randomly attach it to a number of other neurons. The number of neurons to attach to is a randomly chosen number from a Gaussian distribution. We add a synapse to the target neuron, ensuring that there are no loops or multiple edges. We prohibit loops and multiple edges for simplicity and not because of any biological considerations. The synaptic strength is currently set to the same value for all axons. It would not be difficult to modify this so that the link strength is randomly chosen.

2.4 Running the Simulation

We start by selecting a fixed number of neurons at random to be the inputs into the network. Every time step we initiate a signal into a randomly selected subset of these neurons. The pulse propagates throughout the network, occasionally causing neurons to fire. Neuronal regulation occurs at regular intervals, whereas Hebbian learning occurs in conjunction with the firing events. Periodically, we iterate through the list of neurons, initiating a prune action. The prune action eliminates all axons that have a link strength below a certain value. These actions work together to create a self organizing structure out of the initial random graph.

The competition among neurons and synapses can be seen as a measure of fitness. Neurons and synapses that are unfit will have low link strength (in the case of synapses) or few connections to the network (in the case of neurons). These unfit nodes are “detached” from the network. The detachment criteria is a simple threshold in terms of connection strength or number of links. This criteria is determined at the beginning of the simulation and each neuron evaluates its attached synapses and itself. This local criteria could easily be varied to mimic a heterogeneous network.

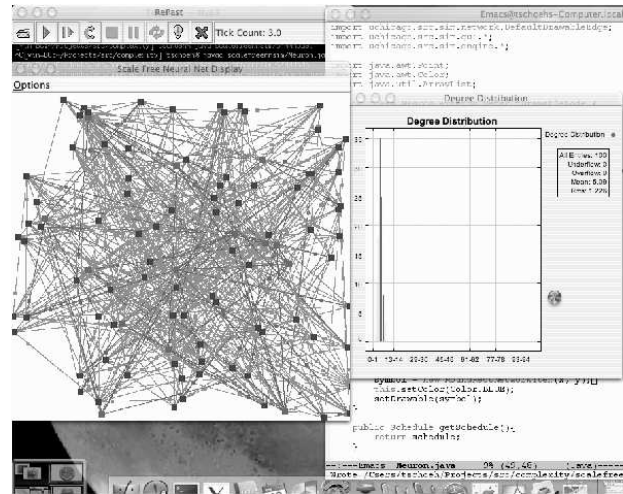


Figure 2: A screen shot of the simulation.

3 Results

Our simulation runs in either a graphical mode or a GUI free batch mode. When in graphical mode, the simulation displays the real time status of the network topology. A side window displays a dynamic histogram of the connectivity of the graph. For a screen shot of the running program, see Figure 2. The batch mode of the simulation runs without the GUI front end. Simulation data, in the form of adjacency matrices and computed information, such as degree distribution, are output as ASCII files.

3.1 Network Structure

We have analyzed the resulting output from our simulations. Using only Hebbian learning, we demonstrate a noticeable change in the network topology. The simulation was run with links initially distributed via a Gaussian distribution with $\mu = 15, \sigma = 1.5$. The simulation was run for 250 time steps, which is more than sufficient for the topology to stabilize. Consider Figure 3, which shows the average clustering coefficient of 5 different runs of the simulation at the same parameter levels. By time step 50, the average clustering coefficient has stabilized, and remains at the same level throughout the simulation.

There are interesting results in terms of link structure as well. Comparing the histograms of link distribution in Figures 4 and 5, we can see a marked change in the link distribution as a result of the running of the simulation. In Figure 5, it is interesting to note the number of neurons with 0 links, suggesting that a result of the link pruning is a reduction in the number of effective neurons in the system.

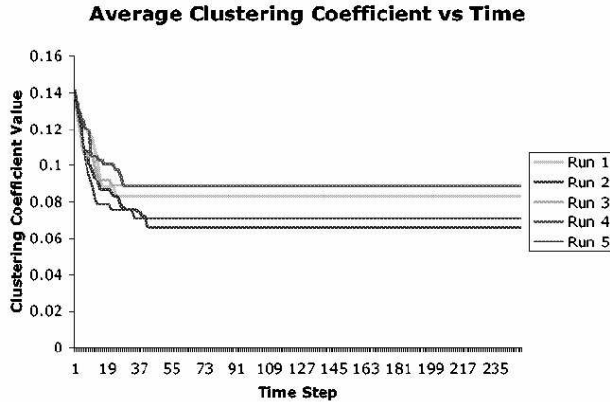


Figure 3: A graph of the average clustering coefficient of 5 runs of the simulation.

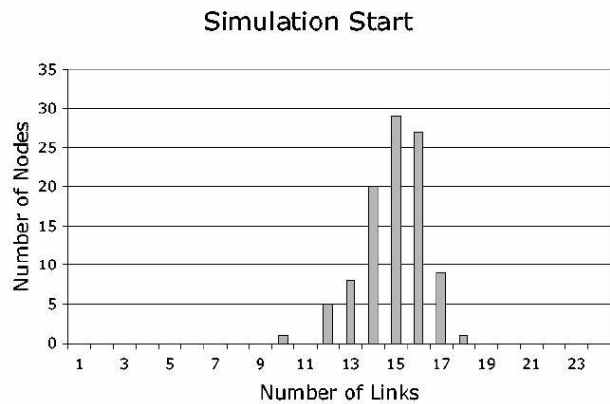


Figure 4: The histogram showing the link distribution at time step 2.

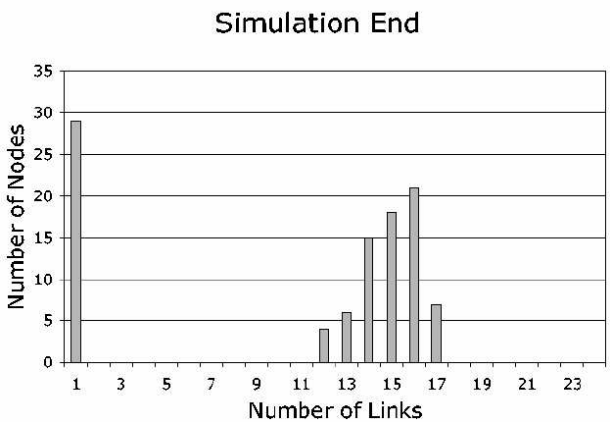


Figure 5: The histogram showing the link distribution at time step 249.

4 Conclusions

Our simulation is an example of the power and flexibility of an agent based approach to modeling. The interaction of a network of neurons, based on local rules and local information, leads to a noticeable global change in the structure of the network. Moreover, once the structure has settled down, it remains constant throughout the remainder of the simulation. This is validation that the swarm programming paradigm is a suitable approach to the modeling of self-organizing neural networks.

A close scrutiny of the link distribution histogram shows that the distribution does not approach the exponential or scale-free distribution that is seen in many complex networks. This histogram describes only one run of the program, yet it is representative of the distributions that were seen for a sweep of the parameter space. It is our belief that pruning in conjunction with Hebbian learning is insufficient to create the complex structure that is seen in biological neural networks.

It is important to note that by the simulation end, in this example, almost 30 neurons have no links. This behavior is very significant in a qualitative way, considering the aforementioned discovery of a significant die off of neurons by adolescence [14]. The trend of the data in this regard is encouraging.

5 Future Work

There is still much progress to be made in biologically inspired self-organizing neural networks. Hebbian learning is clearly an important mechanism, but there are certainly other mechanisms that contribute to the self-organization of biological neural networks.

In the future, we intend to train the network on meaningful patterns in the hopes of benchmarking the network against a more traditional feed-forward network. We also hope to implement the spiking neuron model, which would bring us one step closer to modeling a biological neural network.

References

- [1] Watts, Duncan and S. Strogatz. "Collective Dynamics of 'Small World' Networks" *Nature* **393**, 440-442 (1998).
- [2] Albert, Réka and A.-L. Barabási. "Topology of Evolving Networks: Local Events and Universality." *Physical Review Letters*, Vol 85, No 24. (2000).

- [3] Albert, Réka, H. Jeong, and A.-L. Barabási “Diameter of the World Wide Web” *Nature* **401**, 130-131 (1999).
- [4] Watts, Duncan. *Small Worlds: The Dynamics of Networks Between Order and Randomness* Princeton, NJ: Princeton University Press (1999).
- [5] Barabási, A.-L. *Linked: The New Science of Networks* Cambridge, Massachusetts: Perseus Publishing (2002).
- [6] Harvey, R.L. *Neural Network Principles* Englewood Cliffs, NJ: Prentice Hall, (1994)
- [7] Jeong, H.,B. Tombor, R. Albert, Z. Oltvai and A. Barabási. ”The Large-Scale Organization of Metabolic Networks” *Nature* **407**, 651 (2000).
- [8] Albert, R., H. Jeong and A.L. Barabási. “Error and attack tolerance in complex networks” *Nature* **406**, 378 (2000).
- [9] Anderson, J. “An Introduction to Neural Networks.” Cambridge, MIT Press. (1996).
- [10] McCulloch, W. and Pitts, W. “A Logical Calculus of the Ideas Immanent in Nervous Activity.” *Bulletin of Mathematical Biophysics* **5**, 115-33. (1943)
- [11] Maass, W. “Networks of Spiking Neurons: The Third Generation of Neural Networks.” *Australian Conference on Neural Networks* (1996).
- [12] Hopfield, J.J. and A.V.M. Herz. “Rapid local synchronization of action potentials: Toward computation with coupled integrate-and-fire neurons.” *Proceedings of the National Academy of Science, USA* Vol. 92, pp. 6655-6662 (1995).
- [13] Segev, R. and E. Ben-Jacob. “From Neurons to Brain: Adaptive Self-Wiring of Neurons.” *Advances in Complex Systems*, Vol 1, p67-78. (1998).
- [14] Chalup, Stephan K. “Issues of Neurodevelopment in Biological and Artificial Neural Networks.” *Proceedings of the Fifth Biannual Conference on Artificial Neural Networks and Expert Systems (ANNES 2001)*, pp. 40-45. (2001).
- [15] Chechik, G., I. Meilijson and E. Ruppín. “Neuronal Regulation: A Mechanism For Synaptic Pruning During Brain Maturation.” *Neural Computation*, Vol 11, No 8. (1999).
- [16] Hebb, D.O. *The Organization of Behavior*. Wiley, New York. (1949).
- [17] Gerstner, W. and W.M. Kistler. “Mathematical Formulations of Hebbian Learning.” *Biological Cybernetics*. Vol 87, pp. 404-415. (2002).
- [18] Chechik, G., D. Horn and E. Ruppín. “Neuronal Regulation and Hebbian Learning” In M. Arbib, editor. *The handbook of brain theory and neural networks*. 2nd Edition, MIT Press. (2000).
- [19] Collier, Nick. “RePast: An Extensible Framework for Agent Simulation.” (2003)